

3. Aufgabenblatt

(Besprechung in den Tutorien 08.11.2020–12.11.2020)

Aufgabe 1. Nichtdeterminismus bei Turing-Maschinen

Jede Sprache, die von einer nichtdeterministischen Turing-Maschine akzeptiert wird, kann auch von einer deterministischen Turing-Maschine akzeptiert werden.

Zeigen Sie obige Aussage, indem Sie beschreiben, wie eine beliebige nichtdeterministische Turing-Maschine durch eine deterministische Turing-Maschine simuliert werden kann, sodass beide Maschinen dieselbe Sprache akzeptieren.

Aufgabe 2. LOOP-Programme

1. Seien P_1 und P_2 zwei LOOP-Programme. Simulieren sie das Konstrukt

IF $x_1 > x_2$ **THEN** P_1 **ELSE** P_2 **END**

durch ein LOOP-Programm.

2. Geben Sie ein LOOP-Programm an, das die Funktion $f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ mit $f(x_1, x_2) := x_1 \bmod x_2$ für $x_2 > 0$ berechnet.

Aufgabe 3. LOOP- und WHILE-Programme

1. Warum stoppt jedes LOOP-Programm nach endlicher Zeit?
2. Geben Sie ein WHILE-Programm an, welches niemals stoppt.

Aufgabe 4. Eingeschränkte LOOP-Syntax

Wir betrachten mehrere abgeänderte Formen der Syntax von LOOP. Zeigen oder widerlegen Sie jeweils, dass diese Versionen von LOOP die gleiche Mächtigkeit wie LOOP haben.

1. Anstatt der Zuweisungen $x_i := x_j + c$ und $x_i := x_j - c$ dürfen nur $x_i := x_j + 1$ und $x_i := x_j - 1$ verwendet werden.
2. Anstatt der Zuweisungen $x_i := x_j + c$ und $x_i := x_j - c$ dürfen nur $x_i := x_j + 2$ und $x_i := x_j - 2$ verwendet werden.
3. Anstatt $x_i := x_j + c$ und $x_i := x_j - c$ dürfen nur $x_i := x_j \cdot c$ und $x_i := x_j / c$ verwendet werden. Die Semantik der ersten Anweisung ist dann die Multiplikation und die der zweiten die ganzzahlige Division (sprich: wenn x_j nicht durch c teilbar ist, dann wird das Ergebnis abgerundet).

Aufgabe 5. WHILE- und GOTO-Programme

1. Geben Sie ein GOTO-Programm für die Funktion an, die von folgendem WHILE-Programm (mit Eingaben x_1, x_2) berechnet wird.

```
 $x_3 := x_1 + 1;$   
WHILE  $x_3 \neq 0$  DO  
     $x_2 := x_2 - 1;$   
     $x_3 := x_3 - 1$   
END;  
 $x_0 := x_2 + 0$ 
```

2. Geben Sie ein WHILE-Programm für die Funktion an, die von folgendem GOTO-Programm (mit Eingabe x_1) berechnet wird.

```
 $M_1 :$   $x_2 := x_2 + 1;$   
     $x_0 := x_2 + 0;$   
    IF  $x_1 = 0$  THEN GOTO  $M_3;$   
     $x_1 := x_1 - 1;$   
    GOTO  $M_2;$   
 $M_2 :$   $x_0 := x_0 + 1;$   
    GOTO  $M_1;$   
 $M_3 :$  HALT
```